

Select a Random Sample in Google Sheets

Authored by
Mohammed loot

November 2, 2025

RECOMMENDED CITATION

Mohammed loot (2025). *Select a Random Sample in Google Sheets*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=8374>

In the field of [statistical analysis](#), the ability to extract a truly representative [random sample](#) from a larger population or existing [dataset](#) is fundamentally important. This careful selection process is non-negotiable for ensuring that the results derived from any subsequent analysis are statistically unbiased, robust, and accurately reflective of the characteristics inherent in the entire population. Without proper randomization, analysis risks skewing findings and leading to incorrect conclusions.

Fortunately, modern spreadsheet applications make the process of selecting an unbiased [random sample](#) highly accessible. Specifically, [Google Sheets](#) offers a powerful, yet remarkably simple, built-in mechanism for randomization: the [RAND\(\)](#) function. This function serves as the cornerstone of our methodology, generating a uniformly distributed random decimal number between 0 (inclusive) and 1 (exclusive) with every calculation cycle.

This comprehensive, step-by-step tutorial is designed to guide you through the precise workflow required to leverage the [RAND\(\)](#) function, combined with specific sorting and stabilization techniques, thereby enabling you to select a statistically sound and precise [random sample](#) directly within your spreadsheet environment.

Step 1: Preparing Your Population Dataset for Sampling

The foundational requirement for successful random sampling is ensuring that the source data is correctly organized and structured within the spreadsheet. Before we initiate the randomization process, we must consolidate the values of the complete population or [dataset](#) into a single, contiguous column in [Google Sheets](#). This consolidation ensures that every potential data point is available for equal selection.

For the purposes of this walkthrough, we will assume that we are working with a list of numerical observations located in Column A, beginning in cell A2. This column, which may represent anything from survey responses to financial metrics, constitutes the full population from which the desired sample will be methodically drawn.

	A	B	C	D	E
1	Raw Data				
2	5				
3	6				
4	6				
5	8				
6	10				
7	12				
8	13				
9	13				
10	14				
11	15				
12	17				
13	20				
14	21				
15	22				
16	27				
17					
18					
19					

It is important to note that if your [dataset](#) contains associated identifying labels, timestamps, or multiple variables across adjacent columns, you must ensure that all corresponding rows remain intact. The critical organizational requirement at this stage is that all elements intended for selection are grouped together, ready to be assigned a random key that will keep the entire row associated during the sorting phase.

Step 2: Generating Volatile Random Keys using RAND()

The next critical step involves assigning a unique, temporary random numerical key to every single row within our [dataset](#). This key will dictate the row's position during the subsequent randomization sort. We will designate Column B specifically for this purpose, placing the random key directly adjacent to the original data in Column A.

In cell B2 (the first row corresponding to your data), input the formula: **=RAND()**. Immediately upon entry, the [RAND\(\)](#) function generates a random floating-point number between zero and one. This numerical output will serve as the initial, dynamic randomizing factor for that specific data point.

Once the formula is correctly entered in B2, copy and paste this formula down the entire length of your data range in Column B. This action ensures that every row of your [dataset](#) is now associated with a dedicated, unique random number, thus preparing the entire population for randomization.

	A	B	C	D	E
1	Raw Data	Random Number			
2	5	0.09745187759			
3	6	0.6076916209			
4	6	0.5261692792			
5	8	0.4387673992			
6	10	0.6933042649			
7	12	0.6958931911			
8	13	0.9804370318			
9	13	0.3575659463			
10	14	0.2300584962			
11	15	0.6583574369			
12	17	0.7148310576			
13	20	0.1904448262			
14	21	0.9721330673			
15	22	0.8297472637			
16	27	0.8078631107			
17					
18					
19					
20					
21					

A crucial technical consideration at this point is the nature of the [RAND\(\)](#) function: it is inherently a **volatile function**. This means that its calculated output is designed to change every time the spreadsheet recalculates, which occurs frequently during modifications or system events. If we attempt to sort the data while these volatile formulas are active, the random numbers will shift during the sorting process itself, completely nullifying the intended randomization. Therefore, we must stabilize these values before proceeding.

Step 3: Stabilizing Volatile Formulas using Paste Special

To successfully lock the current random order and prevent the numbers from recalculating during the sorting process, we must convert the active formulas in Column B into fixed, static numerical values. This stabilization process is essential for preserving the transient randomness generated in Step 2.

The stabilization is achieved using the "Paste Values Only" feature:

First, highlight the entire range of calculated, volatile values in Column B (starting from B2). Copy these values using the standard keyboard shortcut: **Ctrl + C** (or Cmd + C on Mac).

Right-click on an empty cell in a temporary column, such as C2, and navigate to the options menu, selecting **Paste special > Paste values only**. This critical step pastes only the fixed numerical results of the formula, discarding the dynamic formula itself.

You may immediately notice that the values in the original Column B change just after the paste operation. This fluctuation is entirely expected because the volatile [RAND\(\)](#) formulas recalculate upon any sheet modification. However, the critical fixed random keys--the numbers we need for sorting--are now safely stored as static data in Column C.

	A	B	C	D	E
1	Raw Data	Random Number			
2	5	0.5565240853	0.09745187759		
3	6	0.07579988487	0.6076916209		
4	6	0.6808403835	0.5261692792		
5	8	0.9249396857	0.4387673992		
6	10	0.4131714107	0.6933042649		
7	12	0.9672858855	0.6958931911		
8	13	0.3586177816	0.9804370318		
9	13	0.4785418919	0.3575659463		
10	14	0.6651376263	0.2300584962		
11	15	0.6641707109	0.6583574369		
12	17	0.2221696466	0.7148310576		
13	20	0.5734229424	0.1904448262		
14	21	0.863426505	0.9721330673		
15	22	0.971635555	0.8297472637		
16	27	0.9551285087	0.8078631107		
17					
18					
19					
20					
21					

Finally, to complete the preparation, overwrite the volatile formulas in Column B. Copy the static values from the temporary Column C and paste them back into Column B. This step ensures that the fixed, non-volatile random key is positioned directly adjacent to the original data in Column A, preparing the entire block for the final sorting stage.

B2		<i>fx</i>	0.0974518775946126			
	A	B	C	D	E	
1	Raw Data	Random Number				
2	5	0.09745187759				
3	6	0.6076916209				
4	6	0.5261692792				
5	8	0.4387673992				
6	10	0.6933042649				
7	12	0.6958931911				
8	13	0.9804370318				
9	13	0.3575659463				
10	14	0.2300584962				
11	15	0.6583574369				
12	17	0.7148310576				
13	20	0.1904448262				
14	21	0.9721330673				
15	22	0.8297472637				
16	27	0.8078631107				
17						
18						
19						
20						

Step 4: Randomizing the Dataset through Sorting by Key

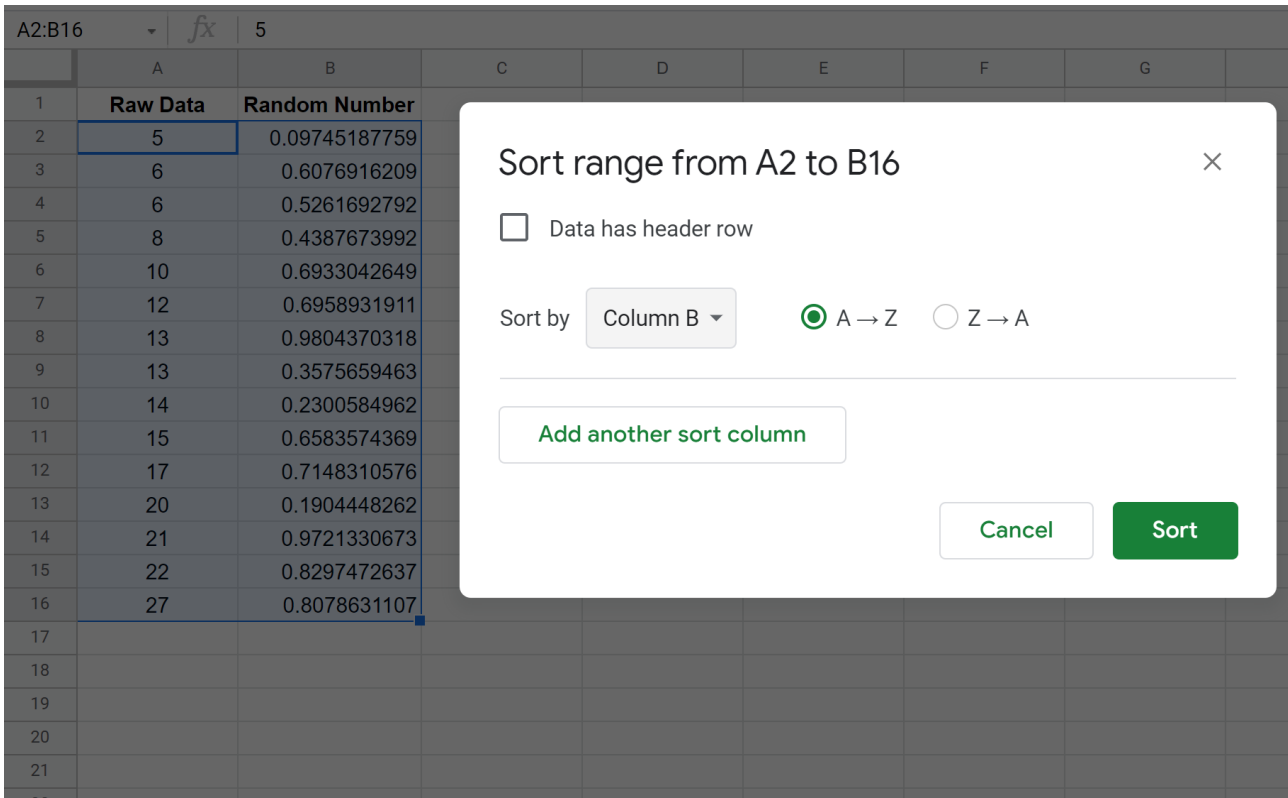
With the static random keys securely assigned and locked to each row, we are now ready to execute the core randomization phase: sorting the entire [dataset](#) based exclusively on these keys. Sorting the data by an arbitrarily generated random number effectively functions as a mechanical shuffle, transforming the original sequential list into a completely randomized order.

Follow these precise steps within [Google Sheets](#) to execute the required sort operation:

Precisely highlight the entire range of your data, ensuring you include both the original data values (Column A) and the static random keys (Column B). In the context of our example, this range would be **A2:B16**.

Locate and click the **Data** tab, which is situated in the main ribbon menu at the top of the [Google Sheets](#) interface.

From the resulting dropdown menu, select the **Sort range** option to initiate the sorting configuration dialog box.



	A	B	C	D	E	F	G
1	Raw Data	Random Number					
2	5	0.09745187759					
3	6	0.6076916209					
4	6	0.5261692792					
5	8	0.4387673992					
6	10	0.6933042649					
7	12	0.6958931911					
8	13	0.9804370318					
9	13	0.3575659463					
10	14	0.2300584962					
11	15	0.6583574369					
12	17	0.7148310576					
13	20	0.1904448262					
14	21	0.9721330673					
15	22	0.8297472637					
16	27	0.8078631107					
17							
18							
19							
20							
21							
22							

Within the sorting dialog, verify that you have configured the sort criteria to order the data specifically by the column containing the static random numbers (Column B). You may choose either ascending (A-Z or smallest to largest) or descending (Z-A or largest to smallest) order. Since the keys themselves are purely random, the direction chosen for the sort is statistically irrelevant to the successful outcome of the overall randomization.

Upon completion of the sort operation, observe that the original data values in Column A have been thoroughly rearranged based on the random key previously assigned and fixed in Column B. Every data point now occupies a statistically random position within the list, making the top rows prime candidates for selection:

A2:B16 *fx* 5

	A	B	C	D	E
1	Raw Data	Random Number			
2	5	0.09745187759			
3	20	0.1904448262			
4	14	0.2300584962			
5	13	0.3575659463			
6	8	0.4387673992			
7	6	0.5261692792			
8	6	0.6076916209			
9	15	0.6583574369			
10	10	0.6933042649			
11	12	0.6958931911			
12	17	0.7148310576			
13	27	0.8078631107			
14	22	0.8297472637			
15	21	0.9721330673			
16	13	0.9804370318			
17					
18					
19					
20					
21					

Step 5: Extracting the Final Random Sample

The process culminates in the final selection phase. First, you must clearly define the required sample size, conventionally denoted as n . Once n is determined, you simply extract the corresponding number of rows from the very top of the newly randomized list. Because the entire population list has been rigorously shuffled based on static random values, the initial n rows selected constitute a perfectly unbiased [random sample](#), ready for analysis.

For example, if your research design dictates the necessity of a random sample of size 5 (i.e., $n=5$), you would proceed to select the first five data values located in Column A immediately following the data sorting process detailed in Step 4.

A2:A6		fx	5			
	A	B	C	D	E	
1	Raw Data	Random Number				
2	5	0.09745187759				
3	20	0.1904448262				
4	14	0.2300584962				
5	13	0.3575659463				
6	8	0.4387673992				
7	6	0.5261692792				
8	6	0.6076916209				
9	15	0.6583574369				
10	10	0.6933042649				
11	12	0.6958931911				
12	17	0.7148310576				
13	27	0.8078631107				
14	22	0.8297472637				
15	21	0.9721330673				
16	13	0.9804370318				
17						
18						
19						
20						

Referring to the visualization above, the resulting [random sample](#) of size 5 would include the values: **5, 20, 14, 13, and 8**. These specific values, representing a non-biased subset of the original population, can then be confidently copied, extracted, and utilized for rigorous statistical modeling, hypothesis testing, or further specialized analysis.

Additional Resources for Advanced Sampling Methods

While the methodology described above provides an elegant and effective solution for simple random sampling within [Google Sheets](#), certain research designs may necessitate more complex sampling schemes. Specialized [statistical software](#) packages (such as R, SPSS, or SAS) offer built-in capabilities for advanced techniques, including stratified sampling, cluster sampling, and systematic sampling, which go beyond the scope of basic spreadsheet randomization.

For users exploring alternative statistical environments or requiring different methodologies, the following tutorials provide guidance on achieving similar random selection goals using specialized platforms: